

SIR GURUDAS MAHAVIDYALAYA
DEPARTMENT OF
Computer Sc.



LAB MANUAL

Subject: Programming in Python
Paper Code: DSE B2

DEPARTMENT OF COMPUTER SCIENCE

S.No.	Topic	Page number
1	write a program to demonstrate different number data types in python(script.py)	
2	write a program to perform different arithmetic operations on numbers in python	
3	write a program to create, concatenate and print a string and accessing sub-string from given string	
4	write a python script to print the current date in the fallowing format”Sun May 29 02:26:23 IST 2017”	
5	write a program to create, append and remove lists in python	
6	write a program to demonstrate working with tuples in python	
7	write a program to demonstrate working with dictionaries in python	
8	program to find the largest number among the three input numbers	
9	Program to convert temperature in Celsius to Fahrenheit	
10	write a python program to construct the following pattern, using a nested for loop	
11	write a python script that prints prime numbers less than 20	
12	write a python program to find the factorial of a number using recursion	
13	write a program that accepts the lengths of three sides of a triangle as input the program output should indicate whether or not the triangle ,is right triangle(recall from the Pythagorean theorem that in a right triangle ,the square of one side equals the sum of the squares of the other two sides)	
14	write a python program to define a module to find Fibonacci numbers and import the module to another program	
15	write a python program to define a module and import a specific function in that module to another program	
16	write a script named copyfile.py. This script should prompt the user for	

	the names of two text files. the contents of the first file should be input and written to the second file	
17	write a program that input a text file .the program should print all of the unique words in the file in alphabetical order.	
18	write a python class to convert an integer to Roman numeral	
19	write a python class to implement pow(x, n)	
20	write a python class to reverse a string word by word.	

1. write a program to demonstrate different number data types in python(script.py)

SOURCE CODE:

```
a = 5
print(a, "is of type", type(a))
a = 2.0
print(a, "is of type", type(a))
a = 1+2j
print(a, "is complex number?", isinstance(1+2j,complex))
output:
```

output:

```
5 is of type <class 'int'>
2.0 is of type <class 'float'>
(1+2j) is complex number? True
```

2: write a program to perform different arithmetic operations on numbers in python

SOURCE CODE:

```
num1 = int(input('Enter First number: '))
num2 = int(input('Enter Second number '))
add = num1 + num2
dif = num1 - num2
mul = num1 * num2
div = num1 / num2
floor_div = num1 // num2
power = num1 ** num2
modulus = num1 % num2
print('Sum of',num1 , 'and' ,num2 , 'is :',add)
print('Difference of',num1 , 'and' ,num2 , 'is :',dif)
print('Product of',num1 , 'and' ,num2 , 'is :',mul)
print('Division of',num1 , 'and' ,num2 , 'is :',div)
print('Floor Division of',num1 , 'and' ,num2 , 'is :',floor_div)
print('Exponent of',num1 , 'and' ,num2 , 'is :',power)
print('Modulus of',num1 , 'and' ,num2 , 'is :',modulus)
```

output:

```
C:\Python34\python.exe G:/Websites/programminginPython/Programs/arithmic_operations.py
Enter First number: 9
Enter Second number 5
Sum of 9 and 5 is : 14
Difference of 9 and 5 is : 4
Product of 9 and 5 is : 45
Division of 9 and 5 is : 1.8
Floor Division of 9 and 5 is : 1
Exponent of 9 and 5 is : 59049|
Modulus of 9 and 5 is : 4

Process finished with exit code 0
```

3: write a program to create, concatenate and print a string and accessing sub-string from given string

SOURCE CODE:

```
create string
# all of the following are
equivalent my_string = 'Hello'
print(my_string)

my_string = "Hello"
print(my_string)

my_string = """Hello"""
print(my_string)

# triple quotes string can extend multiple lines
my_string = """Hello, welcome to
    the world of Python"""
print(my_string)
```

output:

```
Hell
o
Hell
o
Hell
o
Hello, welcome to the world of Python
```

ii) access

string:

ii)concatenate

```
str1 = 'Hello'  
str2 = 'World!'
```

```
# using +  
print('str1 + str2 = ', str1 + str2)
```

```
# using *
print('str1 * 3 =', str1 * 3)
o/p: str1 + str2 = HelloWorld!
str1 * 3 = HelloHelloHello
```

iii) print string:

```
str = 'programiz'
print('str = ', str)
```

```
#first character
print('str[0] = ', str[0])
```

```
#last character
print('str[-1] = ', str[-1])
```

```
#slicing 2nd to 5th character
print('str[1:5] = ', str[1:5])
```

```
#slicing 6th to 2nd last
character print('str[5:-2] = ',
str[5:-2]) output:
```

```
str = programiz
str[0] = p
```

```
str[-1] = z
```

```
str[1:5] = rogr
```

```
str[5:-2] = am
```


4:write a python script to print the current date in the fallowing format”Sun May 29 02:26:23 IST 2017”

SOURCE CODE:

```
# Python Date and Time - Example Program

import time;

localtime = time.asctime(time.localtime(time.time()))

print("The current time = ", localtime);
```

output:

```
>>> ===== RESTART =====
>>>
The current time = Fri Feb 20 09:10:09 2015
>>> |
```

5: write a program to create, append and remove lists in python

SOURCE CODE:

create a list in python.

```
list1 = ['computer', 'programming', 1957, 2070, 3242];  
list2 = [1, 2, 3, 4, 5];  
list3 = ["a", "b", "c", "d", "e"];
```

ex:

```
# Python Lists Example - Creating a list program  
my_list = ["zero", "one", "two", "three"];  
print("Elements of the list, my_list are:");  
for ml in my_list:  
    print(ml);
```

output:

```
>>>  
Elements of the list, my_list are:  
zero  
one  
two  
three  
>>> |
```

ii) Concatenating two lists in python

```
# Lists concatenation in python example  
my_list = ["zero", "one", "two", "three", "four"];  
my_new_list = ["five", "six"];  
my_list += my_new_list;  
print("List's items after concatenating:");  
for l in my_list:  
    print(l);
```

output:

```
>>>
List's items after concatenating:
zero
one
two
three
four
five
six
>>> |
```

iii) To delete any element from a list in python

```
# Deleting element from list in python example
my_list = ["zero", "one", "two", "three", "four"];
print("Elements of the list, my_list are:");
for ml in my_list:
    print(ml);
index = input("\nEnter index no:");
index = int(index);
print("Deleting the element present at index number",index);
del my_list[index];
print("\nNow elements of the list, my_list are:");
for ml in my_list:
    print(ml);
```

output:

```
>>>
Elements of the list, my_list are:
zero
one
two
three
four

Enter index no:|
```

6: write a program to demonstrate working with tuples in python

SOURCE CODE:

Create an empty Tuple in Python

my_tuple = (); here the [variable](#) named my_tuple is the name of the tuple.

Create a Tuple with Items in Python

```
my_tuple = ("me", "my friend", "my brother", "my sister");  
or
```

```
tuple1 = ("python", "tuple", 1952, 2323, 432);  
tuple2 = (1, 2, 3, 4, 5);  
tuple3 = ("a", "b", "c", "d", "e");  
ex:
```

```
# Python Tuple Example  
print("Creating an empty tuple...");  
my_tuple = ();  
print("An empty tuple, my_tuple is created successfully.");  
if not my_tuple:  
    print("The tuple, my_tuple, contains no any item.");  
print("Inserting some items to the tuple...");  
my_tuple = ("me", "my friend", "my brother", "my sister");  
print("\nPrinting the tuple...");  
print(my_tuple);  
print("\nNow printing each item in the tuple...");  
for item_in_tuple in my_tuple:  
    print(item_in_tuple);
```

output:

```
>>>  
Creating an empty tuple...  
An empty tuple, my_tuple is created successfully.  
The tuple, my_tuple, contains no any item.  
Inserting some items to the tuple...  
  
Printing the tuple...  
( 'me', 'my friend', 'my brother', 'my sister' )  
  
Now printing each item in the tuple...  
me  
my friend  
my brother  
my sister  
>>> |
```

7. write a program to demonstrate working with dictionaries in python

SOURCE CODE:

How to create a dictionary

```
# empty dictionary
my_dict = {}
# dictionary with integer keys
my_dict = {1: 'apple', 2: 'ball'}
# dictionary with mixed keys
my_dict = {'name': 'John', 1: [2, 4, 3]}
# using dict()
my_dict = dict({'apple': 1, 'ball': 2})
# from sequence having each item as a pair
my_dict = dict([(1, 'apple'), (2, 'ball')])
```

How to access elements from a dictionary

```
my_dict = {'name': 'Jack', 'age': 26}
# Output: Jack
print(my_dict['name'])
# Output: 26
print(my_dict.get('age'))
)
# Trying to access keys which doesn't exist throws error
# my_dict.get('address')
# my_dict['address']
```

Output : Jack 26

How to change or add elements in a dictionary

```
my_dict = {'name': 'Jack', 'age': 26}
# update value
my_dict['age'] = 27
#Output: {'age': 27, 'name': 'Jack'}
print(my_dict)
# add item
my_dict['address'] = 'Downtown'
# Output: {'address': 'Downtown', 'age': 27, 'name': 'Jack'}
print(my_dict)
```

output: {'age': 27, 'name': 'Jack'}
 {'age': 27, 'address': 'Downtown', 'name': 'Jack'}

How to delete or remove elements from a dictionary

```
# create a dictionary
squares = {1:1, 2:4, 3:9, 4:16, 5:25}

# remove a particular item
# Output: 16
print(squares.pop(4))
# Output: {1: 1, 2: 4, 3: 9, 5: 25}
print(squares)
# remove an arbitrary item
# Output: (1, 1)
print(squares.popitem())
# Output: {2: 4, 3: 9, 5: 25}
print(squares)
# delete a particular item
del squares[5]
# Output: {2: 4, 3: 9}
print(squares)
# remove all items
squares.clear()
# Output: {}
print(squares)
# delete the dictionary itself
del squares
# Throws Error
# print(squares)
```

Output : 16

```
{1: 1, 2: 4, 3: 9, 5: 25}
(1, 1)
{2: 4, 3: 9, 5: 25}
{2: 4, 3: 9}
{}

```

8: program to find the largest number among the three input numbers

SOURCE CODE:

```
# change the values of num1, num2 and num3
# for a different result
num1 = 10
num2 = 14
num3 = 12
# uncomment following lines to take three numbers from user
#num1 = float(input("Enter first number: "))
#num2 = float(input("Enter second number: "))

#num3 = float(input("Enter third number: "))

if (num1 >= num2) and (num1 >= num3):

    largest = num1

elif (num2 >= num1) and (num2 >= num3):
    largest = num2
else:
    largest = num3
print("The largest number between",num1,",",num2,"and",num3,"is",largest)
```

Output : The largest number between 10 , 14 and 12 is 14

9: Program to convert temperature in Celsius to Fahrenheit

SOURCE CODE:

```
# change this value for a different result
celsius = 37.5
# calculate fahrenheit
fahrenheit = (celsius * 1.8) + 32
print('%0.1f degree Celsius is equal to %0.1f degree Fahrenheit' %(celsius,fahrenheit))
```

Output : 37.5 degree Celsius is equal to 99.5 degree Fahrenheit

10:write a python program to construct the following pattern, using a nested for loop

```

#
# #
# # #
# # # #
# # # # #
# # # # # #
# # # # # # #
# # # # # # #
# # # # #
# # # #
# # #
# #
#

```

SOURCE CODE:

```
def run():
    j = 7
    k = 7
    p = 1
    for i in range(8):
        print " " * k,"#" * i
        k -=1
    while j > 1:
        j -= 1
        print " " * p,"#" * j
        p +=1
run()

```

Output :

```

#
# #
# # #
# # # #
# # # # #
# # # # # #
# # # # # # #
# # # # # # #
# # # # #
# # # #
# # #
# #
#

```

11:write a python script that prints prime numbers less than 20

SOURCE CODE:

```
# Python program to display all the prime numbers upto n
# Setting the initial value with 1
Starting_value = 1
# Taking input from the user
n = int(input("Enter the number: "))
print("Prime numbers between", Starting_value, "and", n, "are:")
for num in range(Starting_value, n + 1):
    if num > 1:
        for i in range(2, int(num/2)+1):
            if (num % i) == 0:
                break
        else:
            print(num)
```

Output : Enter the number: 20
Prime numbers between 1 and 20 are:
2
3
5
7
11
13
17
19

12: write a python program to find the factorial of a number using recursion

SOURCE CODE:

```
def recur_factorial(n):
    """Function to return the factorial
    of a number using recursion"""
    if n == 1:
        return n
    else:
        return n*recur_factorial(n-1)

# Change this value for a different result
num = 7

# uncomment to take input from the user
#num = int(input("Enter a number: "))

# check is the number is negative
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    print("The factorial of",num,"is",recur_factorial(num))
```

Output : The factorial of 7 is 5040

13:write a program that accepts the lengths of three sides of a triangle as input the program output should indicate whether or not the triangle ,is right triangle(recall from the Pythagorean theorem that in a right triangle ,the square of one side equals the sum of the squares of the other two sides)

SOURCE CODE:

```
print("Input lengths of the triangle sides:
") x = int(input("x: "))
y = int(input("y: "))
z = int(input("z: "))

if x == y == z:
    print("Equilateral triangle")
elif x==y or y==z or z==x:
    print("isosceles triangle")
else:
    print("Scalene triangle")
```

output:

```
x: 6
y: 8
z: 12
Scalene triangle
```

14: write a python program to define a module to find Fibonacci numbers and import the module to another program

SOURCE CODE:

```
# Fibonacci numbers module

def fib(n):    # write Fibonacci series up to n
a, b = 0, 1
    while a < n:
        print(a, end=' ')
        a, b = b, a+b
print()

def fib2(n): # return Fibonacci series up to n
result = []
a, b = 0, 1
    while a < n:
        result.append(a)
        a, b = b, a+b
    return result

>>> import fibo
```

```
>>> fibo.fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
>>> fibo.fib2(100)
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
>>> fibo._name
'fibo'
```

```
>>> fib = fibo.fib
```

```
>>> fib(500)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```

15: write a python program to define a module and import a specific function in that module to another program

SOURCE CODE:

```
# Python Module example
```

```
def add(a, b):
```

```
    """This program adds two
```

```
    numbers and return the result"""
```

```
    result = a + b
```

```
    return result
```

```
>>> import example
```

```
>>> example.add(4,5.5)
```

```
9.5
```

```
# import statement example
```

```
# to import standard module math
```

```
import math
```

```
print("The value of pi is", math.pi)
```

Output : The value of pi is 3.141592653589793

16: write a script named copyfile.py. This script should prompt the user for the names of two text files. the contents of the first file should be input and written to the second file

SOURCE CODE:

```
#directory: /home/imtiaz/code.py
text_file = open('file.txt','r')
#Another method using full location
text_file2 = open('/home/imtiaz/file.txt','r')
print('First Method')
print(text_file)
print('Second Method')
print(text_file2)
```

The **output** of the following code will be

```
===== RESTART: /home/imtiaz/code.py =====
```

First Method

Second Method

>>>

Python Read File, Python Write File

```
#open the file
text_file = open('/Users/pankaj/abc.txt','r')
#get the list of line
line_list = text_file.readlines();
#for each line from the list, print the line
for line in line_list:
    print(line)
text_file.close() #don't forget to close the file
```

output:

```
>>>
===== RESTART: /Users/pankaj/Desktop/read-file.py =====
Hi Pankaj

I am here

>>>
```

Again, the sample code for writing into file is given below

```
#open the file
text_file = open('/Users/pankaj/file.txt','w')
#initialize an empty list
word_list= [ ]
#iterate 4 times
for i in range (1, 5):
    print("Please enter data: ")
    line = input() #take input
    word_list.append(line) #append to the list
text_file.writelines(word_list) #write 4 words to the file

text_file.close() #don't forget to close the file
```

```
>>>
===== RESTART: /Users/pankaj/Desktop/write-file.py =====
Please enter data:
1
Please enter data:
2
Please enter data:
3
Please enter data:
4
>>>
```

Python Copy File

```
import shutil
shutil.copy2('/Users/pankaj/abc.txt', '/Users/pankaj/abc_copy2.txt')
#another way to copy file
shutil.copyfile('/Users/pankaj/abc.txt', '/Users/pankaj/abc_copyfile.txt')
print("File Done")
```


Python Delete File

We can use below code to delete a file in python.

17: write a program that input a text file .the program should print all of the unique words in the file in alphabetical order.

```
# change this value for a different result
my_str = "Hello this Is an Example With cased letters"

# uncomment to take input from the user
#my_str = input("Enter a string: ")

# breakdown the string into a list of words
words = my_str.split()

# sort the list
words.sort()

# display the sorted words

print("The sorted words are:")
for word in words:
    print(word)
```

o/p:

```
The sorted words are: Example
Hello
Is
```

With an

cased

letters

this

18: write a python class to convert an integer to Roman numeral

Source Code:

```
class py_solution:
    def roman_to_int(self, s):
        rom_val = {'I': 1, 'V': 5, 'X': 10, 'L': 50, 'C': 100, 'D': 500, 'M': 1000}
        int_val = 0
        for i in range(len(s)):
            if i > 0 and rom_val[s[i]] > rom_val[s[i - 1]]:
                int_val += rom_val[s[i]] - 2 * rom_val[s[i - 1]]
            else:
                int_val += rom_val[s[i]]
        return int_val
print(py_solution().roman_to_int('MMMCMXXXVI'))
print(py_solution().roman_to_int('MMMM'))
print(py_solution().roman_to_int('C'))
```

Output :

```
3986
4000
100
```

19: write a python class to implement pow(x, n)

Source Code:

```
# Python3 program to calculate pow(x,n)
```

```
# Function to calculate x
```

```
# raised to the power y
```

```
def power(x, y):
```

```
    if (y == 0): return 1
```

```
    elif (int(y % 2) == 0):
```

```
        return (power(x, int(y / 2)) *  
                power(x, int(y / 2)))
```

```
    else:
```

```
        return (x * power(x, int(y / 2)) *  
                power(x, int(y / 2)))
```

```
# Driver Code
```

```
x = 2; y = 3
```

```
print(power(x, y))
```

Output : 8

20: write a python class to reverse a string word by word.

Source Code:

```
class py_solution:
    def reverse_words(self, s):
        return ' '.join(reversed(s.split()))

print(py_solution().reverse_words('hello .py'))
```

output: `.py hello`